let S = \x y z.x z (y z)
let K = \x y.x
S K K

<whitespace>
→ " " <whitespace_opt>
→ " "

<line_end>
→ <whitespace_opt> ("\n" | "\0")
→ ("\n" | "\0")

<variable>
→ <letter> <variable_opt>
→ <letter>
→ ("S" | "K" | "x" | "y" | "z")

<variables>
→ <variable> <variables_opt>
→ <variable> <whitespace> <variable> <variables_opt>
→ <variable> <whitespace> <variable> <whitespace> <variable> <variables_opt>
→ <variable> <whitespace> <variable> <whitespace> <variable>
→ "x y z"

<term>
→ <application>
→ <operand> <application_opt>
→ <variable> <application_opt>
→ <variable> <whitespace> <operand> <application_opt>
→ "S " <operand> <application_opt>
→ "S " <variable> <application_opt>
→ "S " <variable> <whitespace> <operand> <application_opt>
→ "S " <variable> <whitespace> <operand>
→ "S K " <operand>

→ "S K " <variable>
→ "S K K"

<definition>
→ <declaration> <whitespace> "=" <whitespace> <term>
→ "let" <whitespace> <variable> <whitespace> "=" <whitespace> <term>
→ "let S = " <term>
→ "let S = " <abstraction>
→ "let S = \" <variables> "." <term>
→ "let S = \x y z." <term>
→ "let S = \x y z." <application>
→ "let S = \x y z." <operand> <application_opt>
→ "let S = \x y z." <variable> <application_opt>
→ "let S = \x y z." <variable> <whitespace> <operand> <application_opt>
→ "let S = \x y z.x " <operand> <application_opt>
→ "let S = \x y z.x " <variable> <application_opt>
→ "let S = \x y z.x " <variable> <whitespace> <operand> <application_opt>
→ "let S = \x y z.x " <variable> <whitespace> <operand>
→ "let S = \x y z.x z " <operand>
→ "let S = \x y z.x z " <term_par>
→ "let S = \x y z.x z (" <term> ")"
→ "let S = \x y z.x z (" <application> ")"
→ "let S = \x y z.x z (" <operand> <application_opt> ")"
→ "let S = \x y z.x z (" <operand> <whitespace> <operand> <application_opt> ")"
→ "let S = \x y z.x z (" <operand> <whitespace> <operand> ")"
→ "let S = \x y z.x z (" <variable> <whitespace> <variable> ")"
→ "let S = \x y z.x z (y z)"

<program>
→ <expression> <program>
→ <expression> <expression> <program>
→ <expression> <expression> <expression> <program>
→ <expression> <expression> <expression>
→ <expression_opt> <line_end> <expression> <expression>
→ <definition> <line_end> <expression> <expression>

2

$\rightarrow$ <definition> <line_end> <expression_opt> <line_end> <expression>

$\rightarrow$ <definition> <line_end> <definition> <line_end> <expression>

$\rightarrow$ <definition> <line_end> <definition> <line_end> <expression_opt> <line_end>

$\rightarrow$ <definition> <line_end> <definition> <line_end> <term> <line_end>

$\rightarrow$ "

let S = \x y z.x z (y z)\n

let K = \x y.x\n

S K K\0

"